

⚠ Istruzioni per l'installazione e la configurazione dell'applicazione distribuita "EasyPay" di Andrea Vaiuso

Installazione componenti

L'applicazione distribuita EasyPay consta di 4 componenti principali:

- Server EasyPay
Software necessario: node.js
<https://nodejs.org/it/>
- Database EP
Software necessario: MySql
<https://www.mysql.com/it/>
- Applicazione Web
Software necessario: Browser Web (test effettuati su Google Chrome)
<https://www.google.com/intl/it/chrome/>
- Applicazione Mobile
Software necessario: Smartphone Android o Emulatore con spazio sul disco pari a 11MB
<https://developer.android.com/studio>

Configurazione Server

Per configurare il server è necessario avere installato il framework NODE.JS all'interno del sistema operativo.

Installazione:

1. Copiare la cartella "web-app" All'interno del sistema operativo
2. Linux o mac: Posizionarsi tramite terminale in Progetto\web-app\ e digitare il comando:
npm start
3. Windows: Avviare il file StartServer.bat all'interno della cartella web-app



(IMPORTANTE) Configurazione indirizzi:

Il server e le richieste web sono impostate di default all'indirizzo 127.0.0.1 (localhost) con numero di porta 8001. Si consiglia di modificare questo valore impostando un indirizzo IP statico accessibile fuori dalla LAN. Per fare ciò bisogna:

1. cambiare i valori di "serverAddress" e "port" all'interno del file settings.json situato nella cartella web-app
2. Impostare i valori "addr" e "port" nel file models.js al percorso /web-app/client/assets/js/

Configurazione messaggistica SMS/Whatsapp TWILIO

Per i servizi di messaggistica viene usato il servizio Twilio: <https://www.twilio.com/>

Per configurare il server all'invio di SMS e Whatsapp è necessario inserire numeri di telefono configurati per l'utilizzo di questi servizi (SMS a pagamento, Whatsapp previa abilitazione)

(Purtroppo non mi è stato possibile acquistare tali servizi, o aspettarne l'abilitazione. Il server disabiliterà di default queste funzioni se non vengono configurati numeri compatibili a tali servizi).

Per configurare i numeri basta impostare i valori

```
"TWILIO_ACCOUNT_SID" : "",  
"TWILIO_AUTH_TOKEN" : "",  
"TWILIO_MESSAGING_NUMBER" : "",  
"TWILIO_MESSAGING_WHATSAPP_NUMBER" : ""
```

All'interno del file settings.json situato nel percorso /Progetto/web-app/

Se il server è partito correttamente dovrebbe mostrarsi la seguente schermata:

Configurazione database


Il database funziona tramite il software MySQL. L'account per la gestione del database viene creato automaticamente.

- **Username:** easypay
- **Password:** easypay

Per creare l'account di gestione e le varie tabelle bisogna eseguire i comandi contenuti all'interno del file dbCreation.sql contenuto nella cartella /Progetto/Database/ all'interno dell'ambiente MySQL

Vengono forniti dei dati di prova per poter configurare carte di credito ed account bancari. Inoltre viene fornito un account pre configurato per poter effettuare test di pagamenti (simulati)

Carte (Numero di carta, data scadenza, ccv, Indirizzo di fatturazione, proprietario, disponibilità)

 (Importante) A causa di connessioni non possibili con i veri database bancari vengono simulati i pagamenti tramite le tabelle Bank e Card all'interno del database. Non è dunque possibile registrare carte o account bancari se non vengono prima aggiunte come valide all'interno del database. Per effettuare il test sono state inserite le seguenti carte che dunque possono essere registrate

- Carta 1
- `'4242535364647575','02/22',000,'Via Liguria 7','ANDREA VAIUSO',500`
- Carta 2
- `'5242535364647575','02/22',001,'Viale Delle Scienze 1','ROBERTO PIRRONE',500`

Account bancari: (Iban, proprietario, disponibilità)

- `'IT0E123456789','ANDREA VAIUSO',500`

Account: (email, Nome, Cognome, password, disponibilità, Indirizzo, telefono)

- `'andrea.vaiuso@community.unipa.it','Andrea','Vaiuso','admin',1000.00,'Via Liguria 7','+393345024316'`

Aziende su cui effettuare pagamenti:

- `'NETFLIX'`
- `'AMAZON PRIME'`

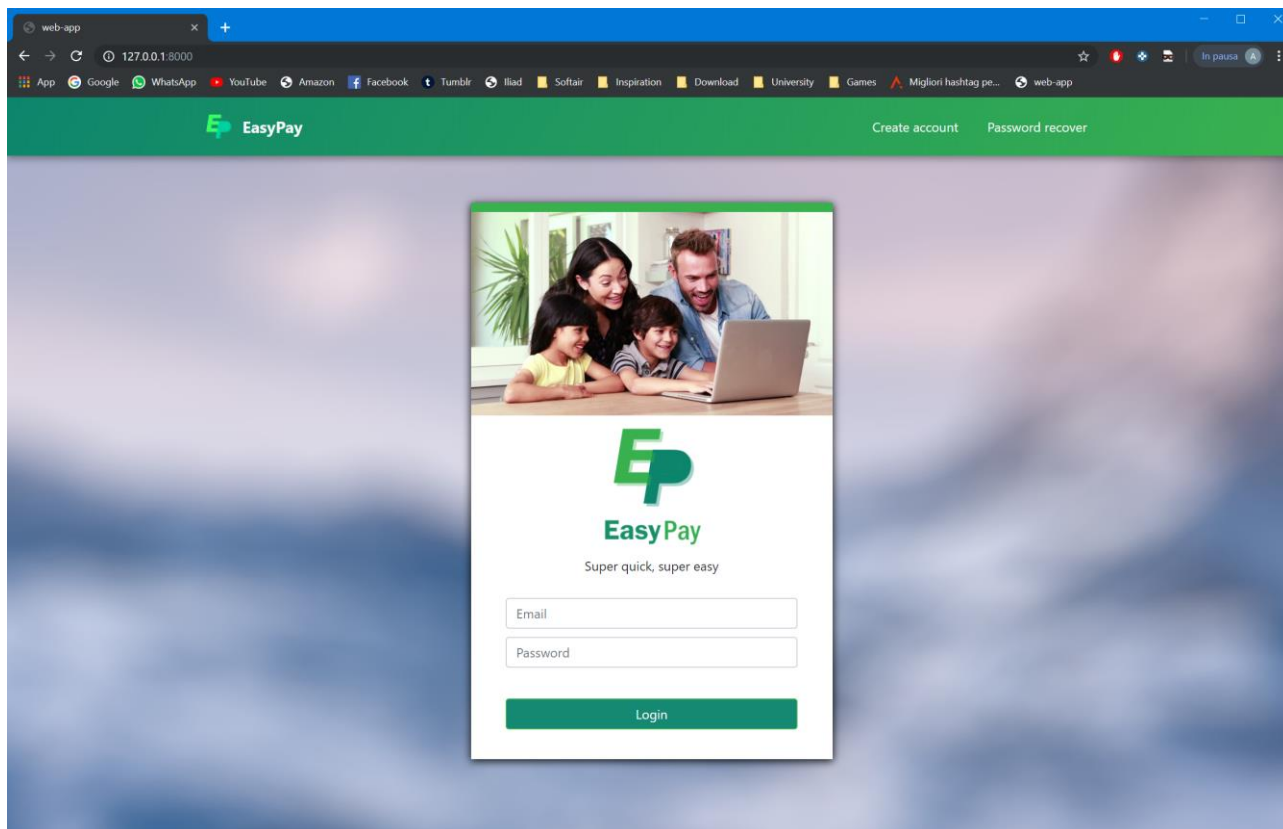
É possibile inserire altri valori direttamente nel database.

Per il test si consiglia di creare un nuovo account tramite la funzionalità "Registrati" e provare i vari metodi di pagamento con l'account predefinito (già attivato) andrea.vaiuso@community.unipa.it

Configurazione web app

Per avviare l'applicazione tramite browser assicurarsi che il server sia in esecuzione e digitare nella barra degli indirizzi `http:// (INDIRIZZO IP SERVER) : (NUMERO PORTA)`

Se il server è avviato correttamente dovrebbe essere mostrata questa pagina:



Configurazione App Mobile

⚠ (Importante) Configurazione indirizzi

Attenzione. Il file APK allegato è compilato in maniera tale da rispondere alle richieste inviate al server situato all'indirizzo IP: **37.116.232.152**

Questo perché non è possibile testare l'applicazione in localhost visto che è necessario un emulatore che non può eseguire il server in locale. Per utilizzare l'applicazione e testarla in ambiente virtuale/fisico con dispositivo esterno, è necessario cambiare l'indirizzo IP a cui vengono effettuate le richieste. Questo valore è contenuto nel file models.js al percorso /Progetto/mobile-app/www/assets/js/

Per compilarlo è necessario creare un file ZIP della cartella mobile-app e caricarlo nel sito Adobe PhoneGap Build: <https://build.phonegap.com/apps>

Se l'applicazione è configurata correttamente dovrebbe essere mostrata questa pagina:

